

**ĐẠI HỌC KHOA HỌC TỰ NHIÊN  
ĐHQG TP HCM  
KHOA CÔNG NGHỆ THÔNG TIN  
LỚP 14CTT2.1**

**BÁO CÁO ĐỒ ÁN  
MÔN KIẾN TRÚC MÁY TÍNH VÀ HỢP NGỮ  
THƯ VIỆN TIME**

**GV: ThS. Vũ Minh Trí**

**Các thành viên:**

|                              |                |
|------------------------------|----------------|
| <b>Nguyễn Thị Thu Hà</b>     | <b>1412140</b> |
| <b>Võ Phương Hòa</b>         | <b>1412192</b> |
| <b>Đoàn Thị Phương Huyền</b> | <b>1412197</b> |

**Tp. Hồ Chí Minh, tháng 11/2015**

## MỤC LỤC

|  |           |
|--|-----------|
| <b>1. Thông tin thành viên .....</b>               | <b>1</b>  |
| <b>2. Yêu cầu đồ án.....</b>                       | <b>1</b>  |
| <b>3. Quá trình làm đồ án .....</b>                | <b>1</b>  |
| <i>a. Chuẩn bị .....</i>                           | <i>1</i>  |
| <i>b. Quá trình cài đặt .....</i>                  | <i>2</i>  |
| <b>4. Quy tắc viết và gọi hàm trong MIPS .....</b> | <b>5</b>  |
| <b>5. Đánh giá kết quả.....</b>                    | <b>6</b>  |
| <b>6. Chạy kiểm tra .....</b>                      | <b>7</b>  |
| <b>7. Nguồn tài liệu tham khảo .....</b>           | <b>10</b> |

## 1. Thông tin thành viên

| Họ và tên             | MSSV    | Email                     |
|-----------------------|---------|---------------------------|
| Nguyễn Thị Thu Hà     | 1412140 | thuhapa@gmail.com         |
| Võ Phương Hòa         | 1412192 | phuonghoa.nt000@gmail.com |
| Đoàn Thị Phương Huyền | 1412197 | dtphuyen2506@gmail.com    |

## 2. Yêu cầu đồ án

- Phải cài đặt đầy đủ các hàm nêu trong phần III dưới dạng thư viện trong MIPS.
- Các hàm thư viện không được sử dụng biến toàn cục để lưu trữ giá trị tính toán trung gian, phải sử dụng biến cục bộ.
- Tuân thủ chặt chẽ quy tắc gọi hàm, mỗi hàm vi phạm quy tắc sẽ bị trừ 50% số điểm của hàm đó.

## 3. Quá trình làm đồ án

### a. Chuẩn bị

- Tìm hiểu cách sử dụng phần mềm MARS.
- Tìm hiểu các quy tắc khi viết và gọi hàm trong MIPS.
- Tìm hiểu về ý nghĩa và cách sử dụng các thanh ghi để tính toán, lưu trữ.
- Tìm hiểu về các lệnh cần thiết cho việc làm đồ án (if..., switch... case..., cách dùng stack).
- Viết chương trình kiểm thử trên Visual Studio bằng ngôn ngữ C++ để tiện cho việc chuyển sang ngôn ngữ MIPS.

*b. Quá trình cài đặt*

**Bảng biểu diễn một số hàm quan trọng  
trong đồ án lập trình hợp ngữ trên MIPS**

| Hàm  | Đối số truyền vào  | Trả về                   | Cách thức cài đặt   |
|--|--|--------------------------|---|
| char* date(int day, int month, int year, char *time) | \$a0: day<br>\$a1: month<br>\$a2: year<br>\$s3: địa chỉ chuỗi time | \$v0: địa chỉ chuỗi time | Stack: lưu trữ các giá trị ngày, tháng, năm và địa chỉ trả về của hàm.<br>Chuyển lần lượt các giá trị ngày, tháng, năm sang dạng chuỗi (bằng hàm convertIntToStr) và thêm vào chuỗi time.<br>(Định dạng chuỗi time: DD/MM/YYYY).  |
| char * convert(char * time, char type)               | \$a0: địa chỉ chuỗi time<br>\$a1: type                             | \$v0: địa chỉ chuỗi time | Dựa vào ký tự type, xác định định dạng cần chuyển:<br>type = 'A' (hoặc 'a'): MM/DD/YYYY<br>type = 'B' (hoặc 'b'): Month DD, YYYY<br>type = 'C' (hoặc 'c'): DD Month, YYYY<br><br>Sử dụng các hàm day(), month(), year() để lấy các giá trị ngày tháng năm (kiểu nguyên).<br><br>Chuyển chuỗi time thành định dạng tương ứng với ký tự type (các hàm sử dụng: convertIntToStr(), strcpy(), strAdd(), strlen()...). |

|                                     |                             |   |  |
|-------------------------------------|-----------------------------|---|--|
| char *<br>weekday(char<br>*time)    | \$a0: địa chỉ<br>chuỗi time | \$v0: địa chỉ<br>chuỗi thứ<br>tương ứng với<br>ngày trong<br>chuỗi time | Sử dụng các hàm day(), month(), year() để lấy<br>các giá trị ngày tháng năm (kiểu nguyên).<br>Tìm $n = (d + m + y + [y/4] + c) \bmod 7$<br>d: ngày trong tháng<br>m: số tương ứng tháng và năm (như file<br>huongdan.pdf)<br>y: hai số cuối của năm<br>[y/4]: phần nguyên của phép chia y/4<br>c: số tương ứng với số dư của phép chia hai số<br>đầu của năm cho 4<br>(0: c = 0; 1: c = 5; 2: c = 3; 3: c = 1)<br>Với n: 0 -> 6 tương ứng với thứ bảy tới thứ sáu. |
| bool leapYear(char<br>*time)        | \$a0: địa chỉ<br>chuỗi time | \$v0: 1 (năm<br>nhuận),<br>0 (năm không<br>nhuận)                       | Lấy năm trong chuỗi time từ hàm year().<br>Kiểm tra năm có nhuận không.<br>(nếu năm chia hết cho 400 hoặc chia hết cho 4<br>nhưng không chia hết cho 100 thì năm nhuận).   |
| int day(char* time)                 | \$a0: địa chỉ<br>chuỗi time | \$v0: giá trị<br>ngày   | Lấy ngày trong chuỗi time dựa vào hàm atoi<br>(chuyển số kiểu chuỗi sang kiểu nguyên).   |
| int month(char*<br>time)            | \$a0: địa chỉ<br>chuỗi time | \$v0: giá trị<br>tháng  | Đưa \$a0 trở tới vị trí ký tự liền sau ký tự '/' đầu<br>tiên. Tại đây, gọi hàm atoi để lấy tháng trong<br>chuỗi time.  |
| int year(char*<br>time)             | \$a0: địa chỉ<br>chuỗi time | \$v0: giá trị<br>năm  | Đưa \$a0 trở tới vị trí ký tự liền sau ký tự '/' thứ<br>hai. Tại đây, gọi hàm atoi để lấy giá trị năm trong<br>chuỗi time.   |
| char *<br>choiceMonth(int<br>month) | \$a0: month                 | \$v0: địa chỉ của<br>chuỗi tháng<br>tương ứng với<br>giá trị month      | Dựa vào tháng nhập vào và tìm chuỗi tháng tiếng<br>anh tương ứng với tháng đó.   |
| char *<br>choiceDay(int<br>num)     | \$a0: num                   | \$v0: địa chỉ<br>chuỗi thứ<br>tương ứng với<br>số num                   | {0, 1, 2, 3, 4, 5, 6} tương ứng với {Saturday,<br>Sunday, Monday, Tuesday, Wednesday,<br>Thursday, Friday}.  |

|   |  |   |  |       |      |               |   |   |   |              |   |                  |   |   |   |       |   |                  |   |
|---|--|---|--|-------|------|---------------|---|---|---|--------------|---|------------------|---|---|---|-------|---|------------------|---|
| int findValue(int month, int leapYear)      | \$a0: month<br>\$a1: {0,1}<br>(leapYear)                         | \$v0: số tương ứng với tháng và năm   | <div>Dựa vào giá trị tháng (month) và leapYear (năm nhuận hay không) thì ta sẽ có kết quả trả về sẽ được lưu ở \$v0.</div> <table><tr><td>Month</td><td>\$v0</td></tr><tr><td>1(\$a1=0), 10</td><td>0</td></tr><tr><td>5</td><td>1</td></tr><tr><td>2(\$a1=1), 8</td><td>2</td></tr><tr><td>2(\$a1=0), 3, 11</td><td>3</td></tr><tr><td>6</td><td>4</td></tr><tr><td>9, 12</td><td>5</td></tr><tr><td>1(\$a1= 1), 4, 7</td><td>6</td></tr></table> | Month | \$v0 | 1(\$a1=0), 10 | 0 | 5 | 1 | 2(\$a1=1), 8 | 2 | 2(\$a1=0), 3, 11 | 3 | 6 | 4 | 9, 12 | 5 | 1(\$a1= 1), 4, 7 | 6 |
| Month                                       | \$v0   |   |  |       |      |               |   |   |   |              |   |                  |   |   |   |       |   |                  |   |
| 1(\$a1=0), 10                               | 0  |   |  |       |      |               |   |   |   |              |   |                  |   |   |   |       |   |                  |   |
| 5   | 1  |   |  |       |      |               |   |   |   |              |   |                  |   |   |   |       |   |                  |   |
| 2(\$a1=1), 8                                | 2  |   |  |       |      |               |   |   |   |              |   |                  |   |   |   |       |   |                  |   |
| 2(\$a1=0), 3, 11                            | 3  |   |  |       |      |               |   |   |   |              |   |                  |   |   |   |       |   |                  |   |
| 6   | 4  |   |  |       |      |               |   |   |   |              |   |                  |   |   |   |       |   |                  |   |
| 9, 12                                       | 5  |   |  |       |      |               |   |   |   |              |   |                  |   |   |   |       |   |                  |   |
| 1(\$a1= 1), 4, 7                            | 6  |   |  |       |      |               |   |   |   |              |   |                  |   |   |   |       |   |                  |   |
| bool checkTime(char *time)                  | \$a0: địa chỉ chuỗi time   | \$v0: 1 nếu chuỗi hợp lệ<br>\$v0: 0 chuỗi không hợp lệ                              | <div>Kiểm tra chuỗi có chứa ký tự khác ký tự '/' và ký tự số hay không.</div> <div>Kiểm tra chuỗi có đủ hai ký tự '/' hay không.</div> <div>Kiểm tra các giá trị ngày, tháng, năm trong chuỗi có hợp lệ hay không.</div> <div>Nếu thỏa cả 3 điều kiện trên, chuỗi hợp lệ, còn lại, chuỗi không hợp lệ.</div>   |       |      |               |   |   |   |              |   |                  |   |   |   |       |   |                  |   |
| bool leapYear_2(int year)                   | \$a0: year   | \$v0 ={1,0}   | <div>Kiểm tra năm có nhuận không.</div> <div>(nếu năm chia hết cho 400 hoặc chia hết cho 4 nhưng không chia hết cho 100 thì năm nhuận).</div>  |       |      |               |   |   |   |              |   |                  |   |   |   |       |   |                  |   |
| int compareDate(char* time_1, char* time_2) | \$a0: địa chỉ của chuỗi time_1<br>\$a1: địa chỉ của chuỗi time_2 | \$v0: 0 (time_1 > time_2)<br>\$v0: 1 (time_1 < time_2)<br>\$v0: 2 (time_1 = time_2) | <div>Sử dụng các hàm day(), month(), year() để lấy các giá trị ngày, tháng, năm từ hai chuỗi.</div> <div>So sánh lần lượt hai năm, hai tháng, rồi hai ngày.</div> <div>Nếu ngày trong chuỗi time_1 nhỏ hơn ngày trong chuỗi time_2 thì trả về 1.</div> <div>Nếu ngày trong chuỗi time_1 lớn hơn ngày trong chuỗi time_2 thì trả về 0.</div> <div>Nếu hai ngày bằng nhau, trả về 2.</div>   |       |      |               |   |   |   |              |   |                  |   |   |   |       |   |                  |   |
| int dayOfMonth(int month)                   | \$a0: month  | \$v0: số ngày của tháng   | <div>Trả về số ngày tương ứng với tháng</div> <div>VD: \$a0 = 1 -&gt; \$v0 = 31</div> <div>\$a0 = 4 -&gt; \$v0 = 30</div>  |       |      |               |   |   |   |              |   |                  |   |   |   |       |   |                  |   |

|  |   |   |  |
|--|---|---|--|
| <pre>int getTime(char* time_1, char* time_2)</pre> | <p>\$a0: địa chỉ chuỗi time_1</p> <p>\$a1: địa chỉ chuỗi time_2</p> | <p>\$v0: khoảng cách giữa hai ngày trong hai chuỗi time_1 và time_2 (đơn vị ngày)</p> | <p>Khai báo biến cục bộ (d1, m1, y1, d2, m2, y2).<br/>So sánh hai chuỗi time_1 và time_2 dựa vào hàm compareDate.<br/>Nếu hai ngày bằng nhau, \$v0 = 0.<br/>Nếu hai ngày khác nhau, gán:<br/>d1, m1, y1: ngày, tháng, năm của ngày nhỏ hơn.<br/>d2, m2, y2: ngày, tháng, năm của ngày lớn hơn.<br/>Kiểm tra:<br/>Nếu y1 == y2: \$v0 = tổng ngày từ d1/m1 -&gt; d2/m2<br/>Nếu y1 &lt; y2: \$v0 = a + b + c<br/>a = tổng ngày từ d1/m1/y1 -&gt; 31/12/y1<br/>b = tổng ngày từ 1/1/y2 -&gt; d2/m2/y2<br/>c = Tổng ngày từ 1/1/y1+1 -&gt; 31/12/y2-1</p> |
|--|---|---|--|

\*Một số hàm khác:

- int strlen(char\* str): tìm độ dài chuỗi.
- char\* strcpy(char\* dest, char\* source): copy chuỗi source vào chuỗi dest.
- char\* strAdd(char\* dest, char\* source): cộng chuỗi source vào cuối chuỗi dest.
- char\* strInsert(char\* str, int pos, char c): chèn ký tự c vào vị trí pos của chuỗi str.
- ...

#### 4. Quy tắc viết và gọi hàm trong MIPS

- Bốn thanh ghi chứa đối số của thủ tục: \$a0, \$a1, \$a2, \$a3 (nếu có nhiều hơn bốn đối số thì các đối số bổ sung được đẩy vào ngăn xếp (stack)).
- Thực hiện 1 lệnh jal hoặc j để chuyển đến các hàm con.
- Kết quả trả về của thủ tục chứa trong thanh ghi \$v0 (và \$v1 nếu cần).
- Phải tuân theo nguyên tắc sử dụng các thanh ghi (register conventions)
  - + \$zero: Luôn bằng 0.
  - + \$s0 - \$s7: (Khôi phục lại nếu thay đổi) Rất quan trọng, nếu thủ tục được gọi (callee) thay đổi các thanh ghi này thì nó phải khôi phục lại giá trị các thanh ghi này trước khi kết thúc.

+ \$sp: (khôi phục lại nếu thay đổi) Thanh ghi con trỏ stack phải có giá trị không đổi trước và sau khi gọi lệnh “jal”, nếu không thủ tục gọi (caller) sẽ không thể quay về được.

+ \$ra: (có thể thay đổi) Khi gọi lệnh “jal” sẽ làm thay đổi giá trị thanh ghi này . Thủ tục gọi (caller) lưu lại (backup) giá trị của thanh ghi \$ra vào stack nếu cần.

+ \$v0, \$v1: (Có thể thay đổi) Chứa kết quả trả về của thủ tục.

+ \$a0, \$a1: (Có thể thay đổi) Chứa đối số của thủ tục.

+ \$t0 -> \$t9: (Có thể thay đổi) Đây là các thanh ghi tạm.

- Nếu thủ tục R gọi thủ tục E:


+ R phải lưu vào stack các thanh ghi tạm có thể bị sử dụng trong E trước khi gọi lệnh jal E (goto E).

+ E phải lưu lại giá trị các thanh ghi lưu trữ (\$s0 - \$s7) nếu nó muốn sử dụng các thanh ghi này -> trước khi kết thúc E sẽ khôi phục lại giá trị của chúng.

+ Thủ tục gọi R (caller) và Thủ tục được gọi E (callee) chỉ cần lưu các thanh ghi tạm (thanh ghi lưu trữ) mà nó muốn dùng, không phải tất cả các thanh ghi!

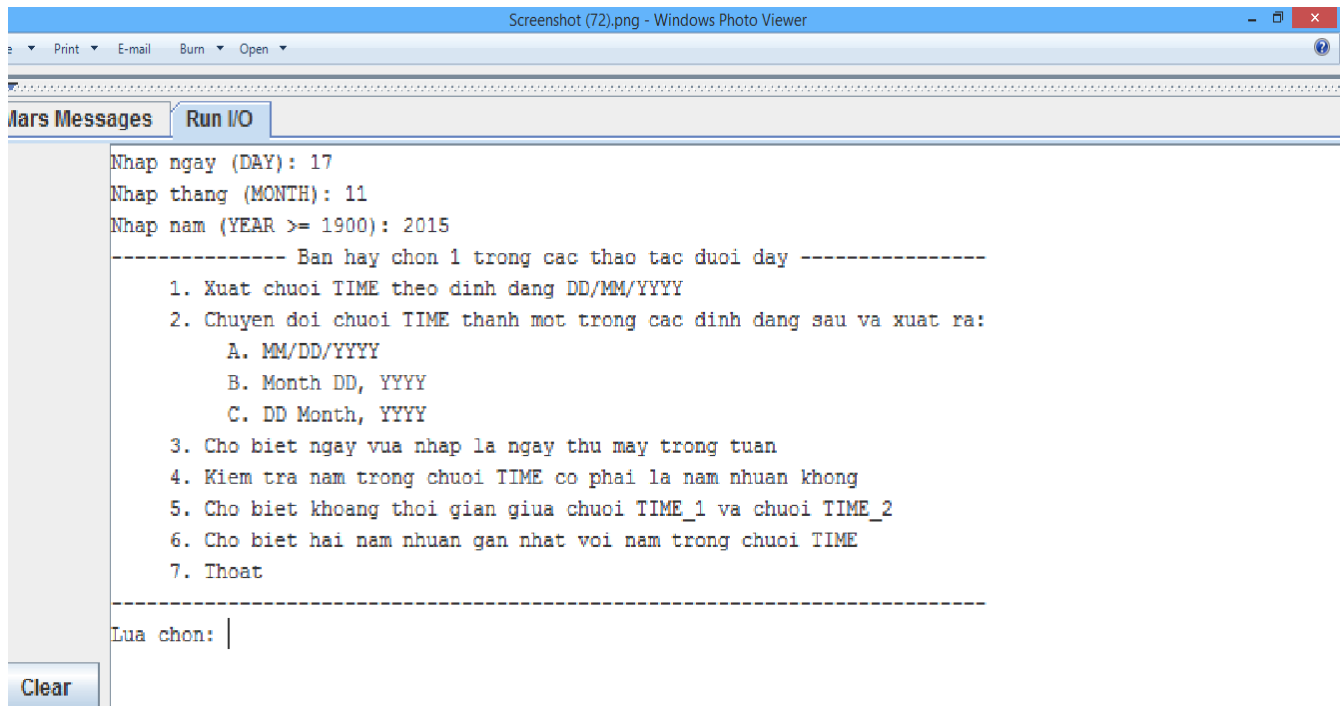
## 5. Đánh giá kết quả

| Chức năng  | Mức độ hoàn thành |
|--|-------------------|
| Cho nhập day, month, year (kiểu nguyên) từ người dùng  | Đã hoàn thành     |
| Xuất chuỗi TIME theo định dạng DD/MM/YYYY  | Đã hoàn thành     |
| Chuyển đổi chuỗi TIME thành một trong các định dạng sau và xuất ra:<br>A. MM/DD/YYYY<br>B. Month DD, YYYY<br>C. DD Month, YYYY | Đã hoàn thành     |
| Cho biết ngày vừa nhập là ngày thứ mấy trong tuần  | Đã hoàn thành     |
| Kiểm tra năm trong chuỗi TIME có phải là năm nhuận   | Đã hoàn thành     |
| Cho biết khoảng thời gian giữa chuỗi TIME_1 và TIME_2 (tính ra số ngày chênh lệch)   | Đã hoàn thành     |
| Chức năng nâng cao: kiểm tra tính hợp lệ của dữ liệu đầu vào   | Đã hoàn thành     |
| Cho biết 2 năm nhuận gần nhất với năm trong chuỗi  | Đã hoàn thành     |
| Thoát  | Đã hoàn thành     |

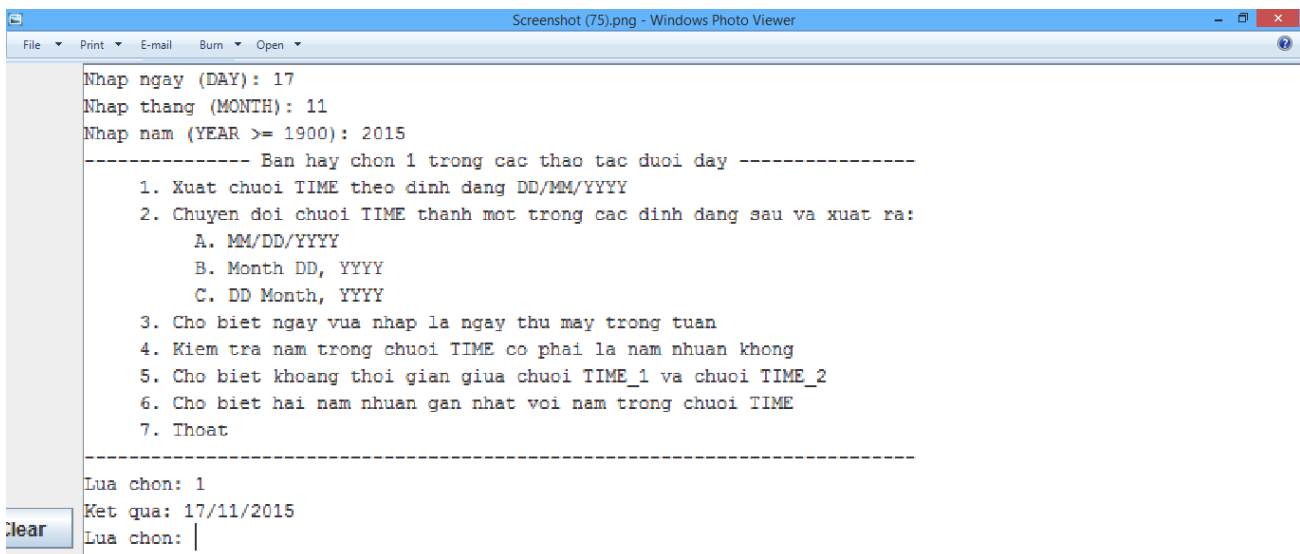
 **Đánh giá chung:** đã hoàn thành 100% đồ án



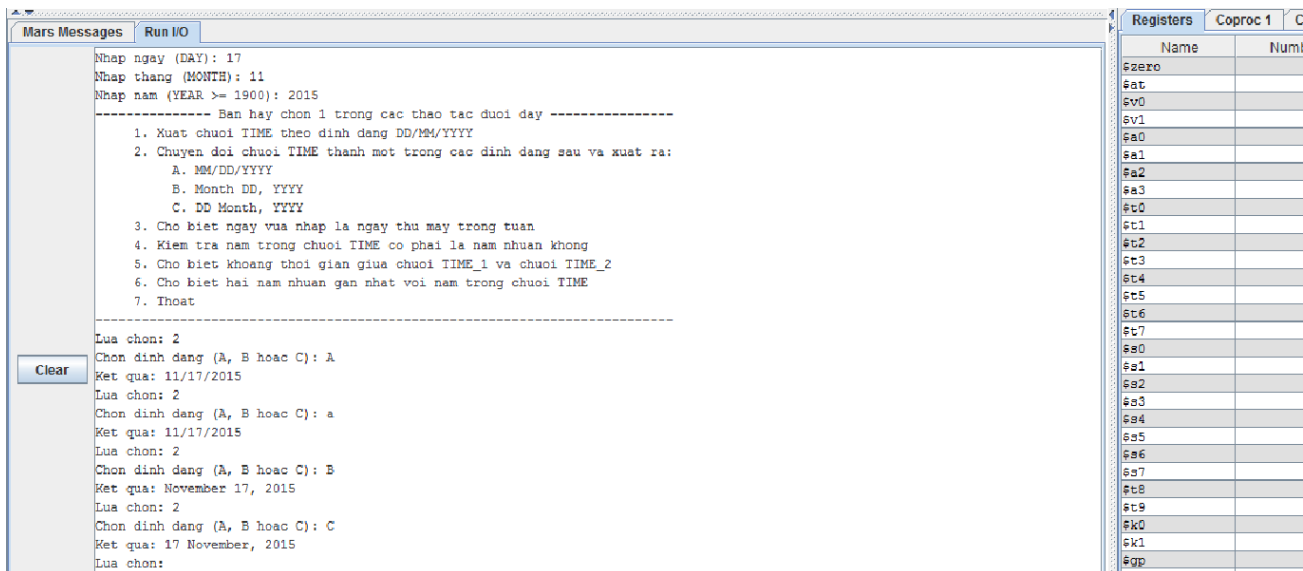
## 6. Chạy kiểm tra



Hình 1.1. Xuất giao diện menu sau khi đã nhập ngày tháng năm

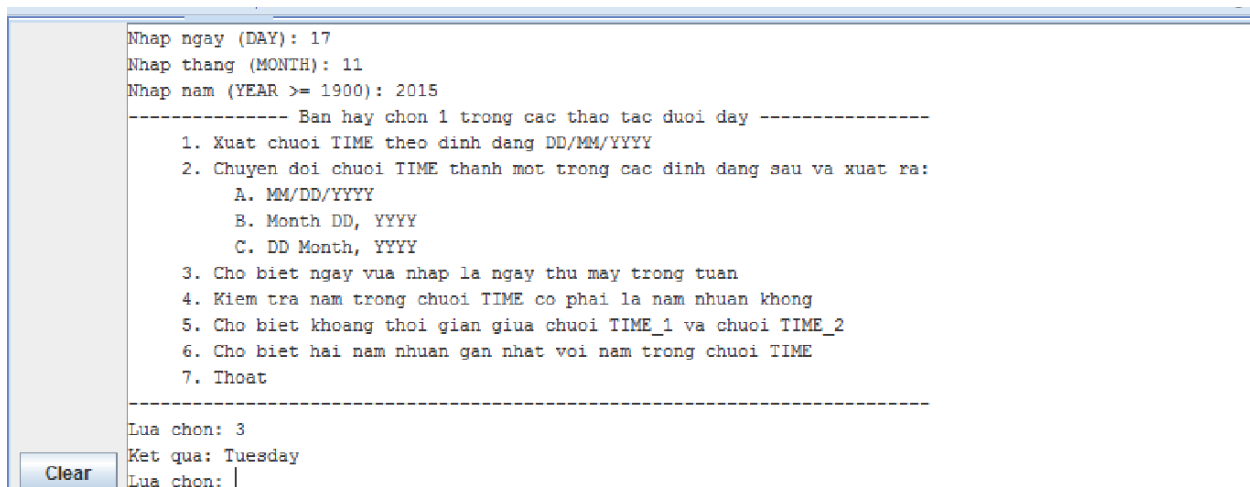


Hình 1.2. Xuất ngày tháng năm theo định dạng DD/MM/YYYY

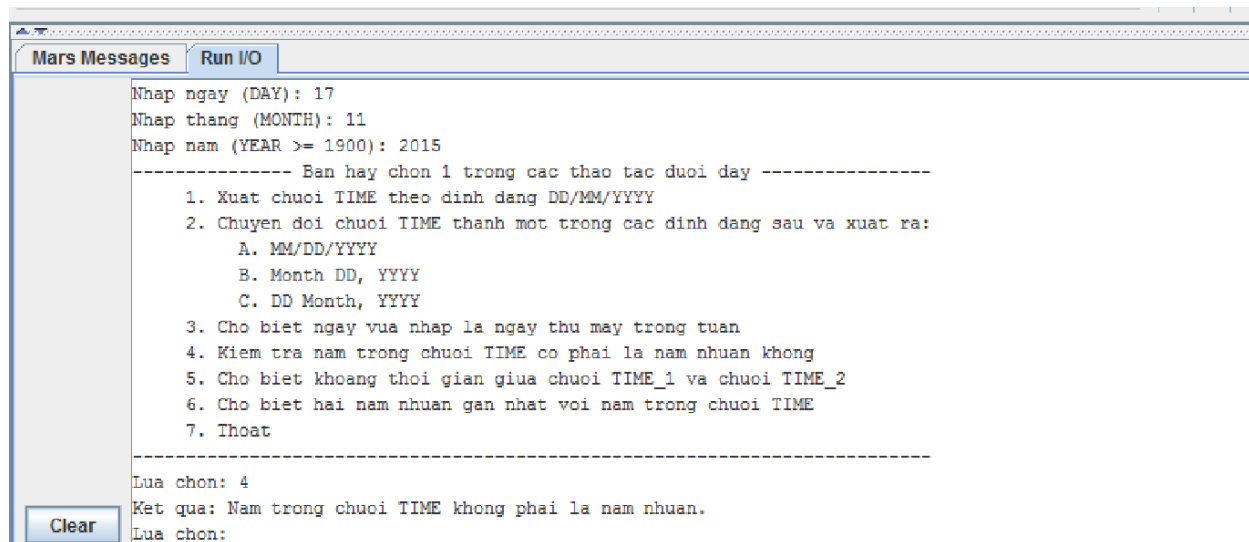


Hình 1.3. Xuất chuỗi time theo định dạng

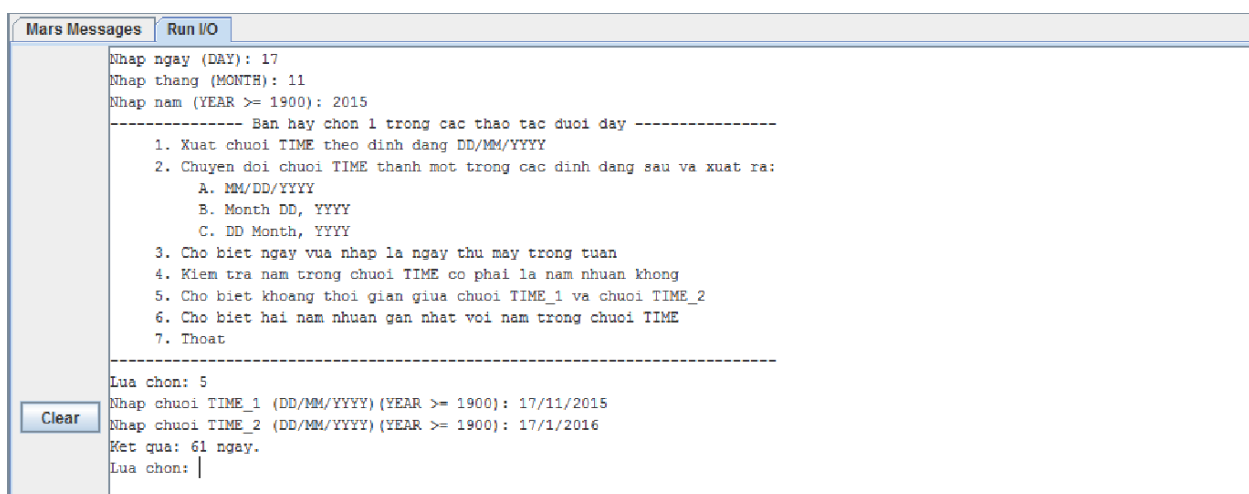
A. MM/DD/YYYY    B. Month DD, YYYY    C. DD Month, YYYY



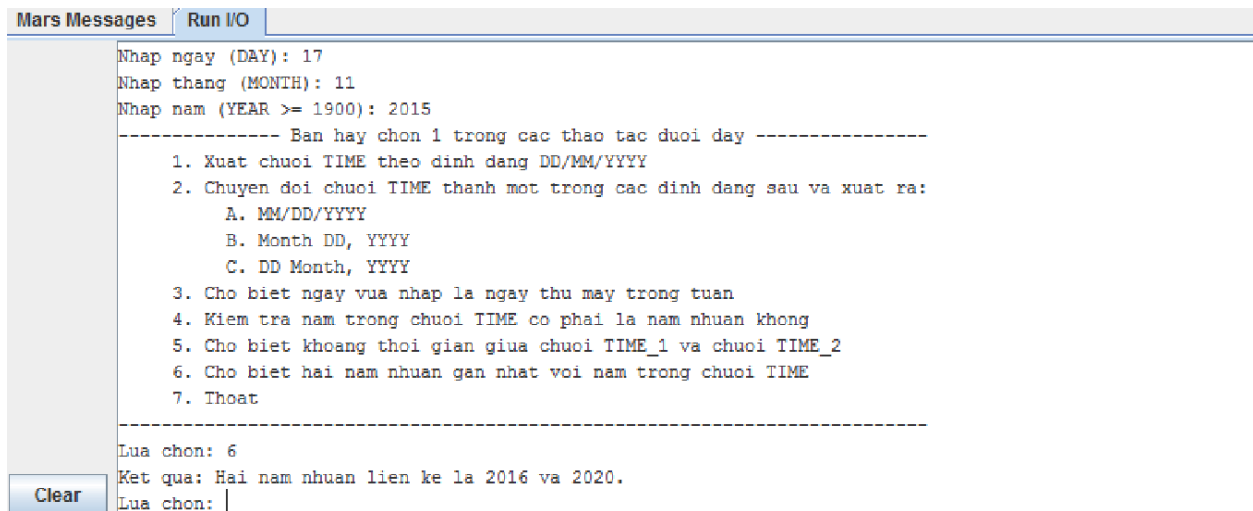
Hình 1.4. Cho biết ngày vừa nhập vào là ngày thứ mấy trong tuần



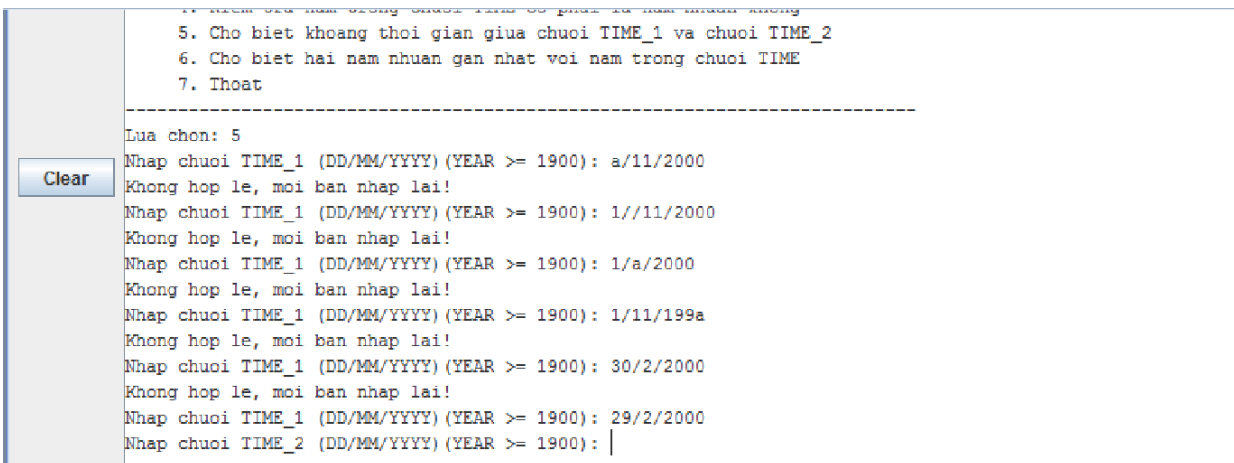
Hình 1.5. Năm trong chuỗi time có phải là năm nhuận hay không.



Hình 1.6. Cho biết khoảng thời gian 2 chuỗi time nhập vào (số ngày chênh lệch).



Hình 1.7. Cho biết hai năm nhuận gần nhất với năm nằm trong chuỗi time



Hình 1.8. Chức năng nâng cao, kiểm tra sự hợp lệ của dữ liệu đầu vào.

## 7. Nguồn tài liệu tham khảo

- Slide lý thuyết của môn kiến trúc máy tính và hợp ngữ.
- MIPS Assembly Language Programming CS50 Discussion and Project Book - Daniel J. Ellard - September, 19.